2. LITERATURE SURVEY

This second chapter provides a detailed survey of the recommender systems, the recommendation techniques and the classification of recommender systems. It discusses in detail about the Multi Criteria Recommender Systems, Multi Criteria based Cloud services selection, recommendation and the challenges involved in modeling recommender systems. It also provides a detailed study about the cold start problem in recommender systems and the approaches available in the literature to solve the cold start problem and discusses about the selected research problem which is the cold start problem in recommending new type of cloud services like the cloud renderfarm services.

2.1 RECOMMENDER SYSTEMS

The explosive growth of E-commerce has led to the need of identifying quickly and accurately, the items or services that a user would be interested in buying online. Recommender systems are such software tools that apply various data mining techniques to identify the items or services that would be of interest to the user [41].

Recommender systems are advantageous to the service providers as well as the users. They help the service providers to predict and identify the right item or product that a user would like to buy based on the user profile. On the other hand, they help the users to compare and evaluate the overwhelming options or alternative items available to them for purchase in the Internet [42, 43]. Recommender systems are popular in various domains like recommending movies, songs, books, travel-destinations etc. For example GroupLens [44] is a recommendation system that helps user to locate relevant news articles from a massive pool of news database. Ringo [45] is an example of an online social information system that builds users profile based on their music album ratings. Amazon, the e-commerce site also constantly upgrades its recommendation algorithms to produce accurate recommendations [46].
In the recent times, the necessity of recommender systems for cloud computing services has also increased as new type of domain specific cloud services are mushrooming in the Internet and the user require recommender systems that can match their requirements to the appropriate type of services. This chapter provides a detailed literature survey about the Recommendation Algorithms used in Recommendation Systems, Multi Criteria Recommender Systems, and the approaches for cloud service selection. The approaches to solve the cold start problem in recommender systems and the recent works in the development of recommendation systems for cloud services have been explored in detail in order to identify the research gap as discussed below in this chapter.

2.2 PHASES OF RECOMMENDATION PROCESS

The phases in the recommendation process include the data collection phase about the user and the services or items as given below in Figure 2.1. The user preferences, learning phase or the modeling phase and the recommendation phase [47]. The three different phases are illustrated in detail below.

- **Data Collection Phase**

In this phase, the domain specific data about the services or items to be recommended to the users are collected using the appropriate data mining techniques and processes like the data cleaning, data stemming, etc. are applied so as to obtain the useful information about the various features of the service or the items. Similarly the data about the user preferences are also obtained from the users. The information about the user preferences can be acquired either explicitly or implicitly.
Generally, recommender systems acquire their user preferences explicitly using the ratings given by the user for a specific item. The rating method opted may be a single rating method where, the user provides a single rating on a scale prescribed about the item. For example a user may rate an item on a scale of 0-10. However, in multi criteria rating method, the user is asked to rate multiple criteria of a single item on a prescribed scale. For example, a hotel recommender system may ask the user to provide rating for multiple criteria like the taste of the food, hygiene, ambience, customer care, cost, etc. on a scale of 0-5 for each criterion and use the ratings for determining the user preferences.

The user preferences can also be acquired implicitly by monitoring the user behavior using his digital footprints on the Internet. These days, as the social media is highly popular, the information about the user in these social media like the items followed, likes, posts, tweets, etc. can be used to identify the user preferences implicitly. The demographic features of users like the location, gender, age, etc. can also be used for this purpose.

Usually, recommender systems collect data about the users from all the different sources to generate knowledge about the users. The knowledge gained is then used to predict the items of user’s interest and new recommendations are generated in the next
interaction with the user. The recommender systems of popular e-commerce sites like Amazon.com and Ebay.com are examples of personalized recommender systems.

- **Learning Phase**

  In the learning phase, the recommendation may use appropriate learning algorithms or methods to learn about the user preferences and about the services using the various information gathered in the previous phase.

- **Recommendation Phase**

  In this phase, the recommender systems predicts the items that may be of interest to the users based on different factors to be considered like the type of feedback, implicit or explicit, source of feedback, user preferences data etc. It then recommends the items to the users based on the recommendation model selected for example a list of top 10 items or a ranked list of items etc. The feedback from the user is again used to learn about the user preferences and all the phases of the recommendation are iterated again.

### 2.3 RECOMMENDATION TECHNIQUES

The selection of the right recommendation technique plays a vital role in determining the quality of the recommendations generated. The classification of the recommendation techniques is given in Table 2.1. There are many works on the taxonomy of the recommendation techniques like the work of Burke et al [48] that classifies the recommendation techniques into five different types, namely the knowledge-based, demographic, utility-based, content-based, collaborative and hybrid recommendation techniques. Montaner et al [49] has proposed an eight-dimensional taxonomy of recommender systems based on two main criteria. The two criteria considered are the generation and maintenance of the user profile and the techniques used to exploit the user profile.

<p>| Table 2.1: Classification of Recommendation Techniques |   |</p>
<table>
<thead>
<tr>
<th>Recommendation Technique</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Knowledge-based recommendation technique</td>
<td>Uses the domain knowledge about the items or services for generating the recommendations [57].</td>
</tr>
<tr>
<td>Demographic recommendation technique</td>
<td>Classifies the users based on their demographic data such as gender, age, education, etc. and recommend the appropriate items [63].</td>
</tr>
<tr>
<td>Utility-based recommendation technique</td>
<td>Computes the utility of each item for a user and recommend the item that has the highest utility value to the users [64].</td>
</tr>
<tr>
<td>Content-based recommendation technique</td>
<td>Recommend the best-matching items by comparing the candidate items with the positively rated items by the user in the past [55].</td>
</tr>
<tr>
<td>Collaborative recommendation technique</td>
<td>Compares the preferences or ratings of other users with similar taste to that of the user to generate recommendations [52].</td>
</tr>
<tr>
<td>Hybrid recommendation technique</td>
<td>Two or more of the above given filtering techniques are combined to overcome some of the drawbacks in using a single filtering system [59], [60].</td>
</tr>
</tbody>
</table>

Another work on the classification of recommendation techniques [50] has omitted the other techniques and has classified the recommendation techniques into three main categories as content-based, collaborative and hybrid recommendation approaches. The recent work [51] emphasizes that the
recommendation technique should be selected based on the type of knowledge available and based on the domain and suggests to analyze the knowledge sources and the type of knowledge that is accessible or available to select the right recommendation technique.

Based on the guidelines provided in this work, the knowledge sources for any recommender system can be broadly categorized into three, namely the Social, Individual user and the Content based knowledge sources. Knowledge about the individual user can be obtained from the user himself in the form of user requirements, opinions, ratings, analyzing user behavior, etc. The Social knowledge sources like the social networking sites (Face book, Twitter, etc.) provide insight about the user preferences implicitly based on the larger community of user with similar taste or preferences like that of the target user. The knowledge about the demographics, requirements, opinions, ratings and user behavior of these larger communities can be used to generate knowledge about the user preferences implicitly. Whereas, the content based knowledge sources include the domain and contextual knowledge. The details about the items to be recommended like the item features, uses, etc. are also used to filter the items that could match the user interests. Based on the type of knowledge sources availability the four popular recommendation techniques applied widely in the recent works include a) Collaborative filtering techniques, b) Content based filtering techniques, c) Knowledge based filtering techniques and d) Hybrid filtering techniques and these techniques are discussed in detail in the following section.

a) Collaborative Filtering Techniques

The collaborative filtering technique compares the preferences or ratings of other users with similar taste to that of the user to generate recommendations [52]. The collaborative filtering techniques are of two types, namely the memory based and the model based collaborative filtering techniques.

The Memory-based filtering algorithms generate predictions using the entire user-item database. They apply statistical methods to find the neighbors or set of users that have similar rating or buying history as that of the target user and use the
knowledge about the preferences of these neighbors to predict the target user items of interest. The user based and item based k Nearest Neighbors (KNN) algorithms are the very, popularly used Memory-based collaborative filtering algorithms. The advantages of memory based filtering algorithms is that they are easy to implement and can incorporate new data easily.

Model-based filtering algorithms build a “model” using the available dataset about the user preferences and use the built "model" to predict the items of interest to the user. The advantages of the model-based filtering approach are the speed, scalability and prediction accuracy [53, 54].

However, the limitations of Collaborative Filtering include a) Cold start problem as the algorithms require huge amount of data to predict the recommendations with accuracy. b) Scalability of the filtering approach may not be easy as it would require huge computational resources. C) The other major disadvantage is the sparsity of data, since not every other user would rate every item that may be of interest to the target user.

b) Content Based Filtering Techniques

Content-based filtering (CBF) algorithms recommend the best-matching items by comparing the candidate items with the positively rated items by the user in the past [55]. The user’s profile that is created based on the information about the user, his preferences in the form of ratings and the item description play an important role in this method. It calculates the similarity of the positive rated item with that of the new items for recommendation. For example, in order to recommend the web pages to the user, the keyword matching method or the Vector Space Model (VSM) are popularly used.

The Vector Space Model (VSM) method creates a spatial representation of the text documents and each document is represented as a n-dimensional space by a vector. Each dimension represented in the model corresponds to a term occurring in the overall vocabulary of the given document. The VSM model calculates the TF-IDF (Term Frequency-Inverse Document Frequency) weighting and infers that if a term occurs frequently in a document (TF =term-frequency) but rarely in the overall vocabulary (IDF
= inverse-document-frequency), then the term is considered to be the relevant topic of the document and the websites based on that term are recommended. For example, if the frequently occurring term is “Movies” then the websites about movies are recommended to the user. The other methods used for estimating the similarity of preferences among the users include the heuristic methods and classification algorithms like nearest neighbors methods, rule induction, linear classifiers, Rocchio’s algorithm, and probabilistic methods [56].

c) **Knowledge Based Filtering Techniques**

Knowledge based filtering techniques use the domain knowledge about the items or services for generating the recommendations. The important techniques applied to domain knowledge representation are the taxonomy and the ontology of the domain items. Ontology is preferred more than the taxonomy as the ontology also captures the relationship between the domain attributes and the items and captures a deeper domain knowledge compared to ontology. Examples of knowledge based recommender systems include the Entree [57] that uses ontology of cuisine to recommend the restaurants. Similarly ontology about the financial domain can be used to recommend the financial products to the user [58]. The advantages of the knowledge based filtering are that they can be used to overcome the cold start problem when the recommender has no idea about the user preferences and use the domain knowledge to recommend the items. The disadvantage of using this method is that it is more expensive as it requires knowledge engineering and continuing maintenance.

d) **Hybrid Filtering Techniques**

Two or more of the above discussed filtering techniques are combined to overcome some of the drawbacks in using a single filtering system and the combination is called the hybrid filtering techniques. The hybrid filtering techniques have been proposed in various combinations based on the source of knowledge and the challenges to be overcome. For example the knowledge based filtering technique can be combined with the content based or collaborative filtering techniques to overcome
the cold start problem in recommender systems or the collaborative filtering technique can be combined with the content based filtering as in the case of Netflix that recommends movies [59], [60].

2.4 CLASSIFICATION OF RECOMMENDER SYSTEMS

![Classification of Recommender Systems](image)

Figure 2.2 Classifications of Recommender Systems

The classification of the Recommender systems is given in Figure 2.2. Generally the Recommender systems are classified as the Memory-based, Model-based, Knowledge-based, Multi-Criteria Recommender systems and hybrid Recommender systems.
The Recommender systems can be built for generating both, non – personalized and personalized recommendations. Non – personalized recommenders are generally easier to build and have limited scope for research.

Recommender systems that recommend the top ten items like songs or books, etc. are examples of non– personalized recommenders. The personalized recommender systems try to predict the suitable items or products for users based on their personal preferences and usually provide a ranked list of recommended items.

The following sessions explains some of the popularly used Recommender systems in detail:

i. **Content based recommender systems** [61], these are the systems that predict a user’s interest for a new item based on the items that the user has liked in the past.

ii. **Collaborative recommender systems**, predict the interest of a user for a new item based on the preferences of the other user with similar taste and preferences as that of the user [62].

iii. **The Demographic recommender systems** are the systems that classify the users based on their demographic data such as gender, age, education, etc. and recommend the appropriate items [63].

iv. **Utility based recommendation systems**, compute the utility of each item for a user and recommend the items that has the highest utility value to the users [64].

v. **Knowledge based recommender systems**, uses logical inferences obtained from the knowledge representations about the user preferences and the item characteristics to suggest items to the users [65].

vi. **Multi criteria recommenders**, define the recommendation of items problem as a multi criteria decision making problem and apply the Multi Criteria Decision
Making methods (MCDM) methods to rank the items based on the importance and suggest the ranked list of items as the recommendations to the users [66].

vii. Finally, the **Hybrid recommender systems** [67], are the systems that combine two or more of the above discussed recommendation methods to improve the accuracy of the recommendations made to the users.

## 2.5 MULTI CRITERIA RECOMMENDER SYSTEMS

Multi Criteria Recommender systems evaluate multiple attributes or criteria to determine whether an item would be of interest to the users for recommending the items or services [68,69]. These systems apply the Multi Criteria Decision Making (MCDM) methods to recommender systems to facilitate the recommendation process. Thus, in these types of recommender systems, the recommendation problem is modeled and solved as an MCDM problem [68].

For solving the MCDM problem of recommending services or items, the Roy’s General modeling methodology suggested for decision making problems is used [69]. This method includes four steps. The first step is to define the “Object of Decision”, based on which the decision has to be made. The second step is to identify and model a set of criteria that could affect the recommendation decisions as “Family of Criteria”. The third step is to define the global preference function to aggregate all the marginal preferences about each item into the global preference of the decision maker about each service or item considered for recommendation. All these four steps help to identify the MCDM dimensions involved in a multi-criteria recommender system [70, 71].

### 2.5.1 Model Multi Criteria Recommender Systems

The steps involved in modelling the Multi Criteria Recommender Systems is given below in Figure 2.3.
Multi criteria recommendations systems define the recommendation problem as a multi criteria decision making problem and apply the Multi Criteria Decision Making (MCDM) methods to rank the items based on the importance and suggest the ranked list of items as the recommendations to the users [68].
The generic methodology for modeling and analyzing the recommendation problem as a decision making problem, involves the following four steps as given below [69].

**Step 1: To define the object of decision making**

In this first step, the set of alternatives or items that needs to be compared has to be decided and defined.

**Step 2: To define a consistent family of criteria**

In this step, the functions or the attributes of the items that expresses the preferences of the decision maker that would have an impact on the decision made by the user is identified and defined.

**Step 3: To develop a global preference model**

In this step, a function is defined in such a way, that it synthesizes the partial preference of each criterion or alternative involved in a model that represents the total preferences of the user regarding an alternative.

**Step 4: To select the decision support process**

This final step is about deciding on the design and development of the procedures, methods and software systems that can provide support to the decision maker in deciding about the right alternative or item.
2.5.2 **Types of Multi-Criteria Recommender Systems**

The multi criteria recommender systems are categorized generally based on a) the decision problematic supported by the recommender system. For example decision problematic supported may be of ranking the alternatives, sorting the alternatives and rarely the choice and description problematic are also used. b) the global preference modeling approach followed like the value-focused models, optimization models, outranking relations and other preference models. Finally, c) the type of criteria for example the type of criteria may be measurable, ordinal, fuzzy and probabilistic. Most of the recommender systems in the literature focus on the decision problematic category of ranking the alternatives and uses the popular MCDM methods like AHP, TOPSIS etc. for ranking the alternatives [66].

2.5.3 **Multi Criteria Rating for Recommendation**

The multi criteria rating based recommendations are gaining popularity compared to the traditional single criteria rating based recommendations due to the increased level of accuracy of recommendations generated. In the multi criteria rating systems, multiple criteria about the same item to be recommended is rated by the user. For example, a restaurant recommendation system, prompts the user to rate multiple criteria like the ambience, hygiene, taste of the food, cost, etc. and these additional information about the user preferences are used to improve the accuracy of the predictions. Multi criteria ratings can be used during the prediction, recommendation stages of the recommendation phase or they could be used to filter recommendations. In the above stated example, the user may want to recommend the restaurants that have scored the highest in ‘taste’ criteria irrespective of the rating scored in the other criteria [66].

2.5.4 **MCDM Methods for Multi Criteria Selection and Recommendation**
<table>
<thead>
<tr>
<th>Author Name</th>
<th>MCDM Methods for Multi Criteria Selection and Recommendation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lee et al [74]</td>
<td>Integrated AHP and data mining methods</td>
</tr>
<tr>
<td>Fasanghari et al [76]</td>
<td>Fuzzy Delphi method</td>
</tr>
<tr>
<td>Palanivel et al [77]</td>
<td>Fuzzy Multicriteria Decision-Making Approach (FMCDM) with Collaborative filtering techniques.</td>
</tr>
<tr>
<td>Huang, Shiu-Li et al [78]</td>
<td>Multi-Attribute Utility Theory (MAUT)</td>
</tr>
<tr>
<td>Chen, Deng-Neng, et al [79]</td>
<td>AHP method</td>
</tr>
<tr>
<td>Vahid et al [81]</td>
<td>ELECTRE (Elimination and Choice Translating Reality), PROMETHEE, Simple Additive Weighting (SAW), Weighted Sum Model (WSM), Weighted Product Model (WPM), TOPSIS (Technique for Order Preference by Similarity to Ideal Solution)</td>
</tr>
</tbody>
</table>

Many have incorporated the MCDM methods in recommender systems as given in Table 2.2. Lakiotaki et al has developed a recommender system based on multiple criteria analysis called the Uta-rec [72].
Lee et al have worked towards developing an agent-based decision making recommender system for the electronic marketplace [73]. Liu et al has integrated AHP and data mining methods for recommending products based on the customer lifetime value [74]. Roland et al [75] has developed a Context-aware Point Of Interest (POI) recommendation system using multi-criteria decision making methods to recommend gas stations to the car drivers. Fasanghari et al [76] has designed a fuzzy expert system using the fuzzy Delphi method for Tehran Stock Exchange to recommend the stocks for portfolio creation. Palanivel et al [77] has combined the Fuzzy Multicriteria Decision-Making Approach (FMCDM) with Collaborative filtering techniques to rank the items and generate recommendations. Huang, Shiu-Li et al [78] has explored using the multi-attribute utility theory (MAUT) MCDM methods to design utility-based recommender systems for e-commerce. Chen, Deng-Neng, et al. [79] has used the AHP method to develop a personalized recommender for mobile phone selection.

Thus, it is evident that the MDCM have been popularly used in designing the recommender systems and a survey on the MCDM methods has been done by Evangelos et al [80] and taxonomy of the MCDM methods used for the selection of the web services has been proposed by the Vahid et al [81]. From these works, it could be concluded that the popular MCDM methods used for services selection and recommendations include AHP (Analytic Hierarchy Process), ELECTRE (Elimination and Choice Translating Reality), PROMETHEE, Simple Additive Weighting (SAW), Weighted Sum Model (WSM), Weighted Product Model (WPM), TOPSIS (Technique for Order Preference by Similarity to Ideal Solution) etc.

### 2.6 CLOUD SERVICES SELECTION AND RECOMMENDATION

#### 2.6.1 MCDM Based Ranking and Selection of Cloud Services

Many works have applied MCDM methods for ranking and selecting the cloud services. Some of the important related works are discussed in this section.
Buyya et al [82] has proposed the SMI Cloud to appraise and select the cloud services based on multiple criteria requirements of the cloud users. The Analytical Hierarchical Process (AHP) method of MCDM ranking mechanism has been applied to identify the right service provider. The other closely related work is the work of Tran et al. [83] on ranking the web services. Where, it applies the Analytical Hierarchical Process (AHP) for ranking web services which have different characteristics from that of the cloud services and user requirements.

Saravanan et al [84] has proposed a framework for ranking based on QoS attributes and advanced reservation of cloud services. However, only few QoS attribute specific to the IaaS type of cloud services has been considered. Hamzeh Khazaeei et al [85] uses an approximate Markov chain model for evaluating the performance of Cloud computing Center. However, this work considers only the response time as the major factor. Ani [86,87] considers QoS constraints like budget, deadline, penalty rate ratio, etc. for selection of cloud resources. Zibin Zheng et al. [88] has developed a predictive framework for Cloud services that predicts the service rank based on the past QoS experiences of user using collaborative filtering methods. However, the QoE attributes have not been considered. Many other works have also applied MCDM methods like DEA, TOPSIS, SAW etc. for comparing and ranking the cloud services based on the QoS requirements [89].

2.6.2 Recommender Systems for Cloud Services

Many research works have focused on developing the recommender system for cloud services as given in Table 2.3. Kailash Nathan et al [90], has developed a social recommender that uses cloud settings for collaborative filtering and for recommending items using the clustering and the fusion technique of collaborative filtering. However, it has not addressed the cold start problem in recommending cloud services that arises in all the three situations that were discussed earlier.

The CSRecommender [91] is a search engine and a recommender system for the discovery of cloud services for a variety of e-commerce genre from news to general information. It applies a hybrid approach of both content-based and collaborative
filtering for recommendation. However, it has considered only the general domain and has not considered the research issues in recommending domain specific cloud services.

Table 2.3: Recommender Systems for Cloud Services

<table>
<thead>
<tr>
<th>Author</th>
<th>Year</th>
<th>Algorithm / Method</th>
<th>Domain</th>
<th>Limitation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kailash Nathan et al</td>
<td>2016</td>
<td>Clustering and fusion technique of Collaborative Filtering.</td>
<td>SaaS</td>
<td>Does not address core Cold start problem.</td>
</tr>
<tr>
<td>John Wheal et al</td>
<td>2015</td>
<td>Uses a hybrid approach of both content-based and Collaborative Filtering.</td>
<td>All types of cloud services.</td>
<td>Business to customer type of general recommender.</td>
</tr>
<tr>
<td>Irfan, Rizwan et al</td>
<td>2015</td>
<td>Uses Weighted Sum Approach (WSA)</td>
<td>Venue Recommender</td>
<td>Uses cloud for only storing the data.</td>
</tr>
<tr>
<td>Q. Yu et al</td>
<td>2014</td>
<td>Matrix tri-factorization, Collaborative filtering</td>
<td>E-Com</td>
<td>Considers only QoS attributes. Does not address core cold start problem.</td>
</tr>
<tr>
<td>Nandini et al</td>
<td>2014</td>
<td>Collaborative filtering</td>
<td>Smart Phones</td>
<td>Does not address core cold start problem in Collaborative filtering methods.</td>
</tr>
<tr>
<td>Ronen, Royi, et al</td>
<td>2013</td>
<td>Uses a hybrid method.</td>
<td>E-Com</td>
<td>Considers only positive observations to filter services.</td>
</tr>
<tr>
<td>Jung et al</td>
<td>2013</td>
<td>Filters services based on QoS requirements.</td>
<td>IaaS cloud services</td>
<td>Considers only the QoS and not the QoE</td>
</tr>
<tr>
<td>Author</td>
<td>Year</td>
<td>Algorithm / Method</td>
<td>Domain</td>
<td>Limitation</td>
</tr>
<tr>
<td>-------------------</td>
<td>------</td>
<td>---------------------------------------------</td>
<td>-------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Zhang et al [98]</td>
<td>2013</td>
<td>Similarity Reasoning</td>
<td>General cloud services.</td>
<td>Does not address core cold start problem. Considers only the QoS and not the QoE attributes.</td>
</tr>
<tr>
<td>Leony, Derick et al [99]</td>
<td>2012</td>
<td>Collaborative Filtering Learning Resource</td>
<td>IaaS cloud services</td>
<td>Uses cloud for only storing the data. Does not address core cold start problem in Collaborative filtering methods.</td>
</tr>
<tr>
<td>Zhang, Miranda et al [100]</td>
<td>2012</td>
<td>Declarative logic based language, and relational data model.</td>
<td>IaaS cloud services</td>
<td>Considers only the QoS and not the QoE attributes. IaaS domain specific.</td>
</tr>
</tbody>
</table>

MobiContext [92], is a context-aware cloud-based recommendation framework, that applies the hybrid cloud-based Bi-Objective Recommendation Framework (BORF) for mobile social networks that suggests users about a venue. Though it uses the Weighted Sum Approach (WSA), it uses cloud for the purpose of storing the data and does not address the research issues in recommending cloud services.

CFSF [93] is a cloud-based recommender for large-scale E-commerce. It dynamically constructs a locally-reduced item-user matrix of the active user item by selecting the top M similar items and top K like-minded users from user clusters. The soothing and fusion algorithm, the Collaborative filtering techniques have been applied in this work. However, even in this work, the cloud storage services have been used for the purpose of storing the data and does not address the core cold start problem.
Cloudrec [94] is a framework for personalized service recommendation in the cloud. It applies the matrix tri-factorization and collaborative filtering techniques on a community-based model to discover a set of homogenous user and service communities from scarce and large-scale QoS data and decides about offloading compute intensive applications to cloud. However, this work considers only the QoS attributes for recommendation and also has not addressed the cold start problem in recommending the cloud services.

U Nandini et al [95] has proposed a mobile recommendation engine that collects context and client sensitive information from smart phone users, However, this work has not addressed the core cold start problem in collaborative filtering methods.

Roiy et al [96] proposes the recommender engine as a cloud service in this work and applies a hybrid method to provide easy-to-use interfaces that enables to integrate a recommendation service into any E-Commerce website. However, this work considers only the positive observations to filter services.

Cloudadvisor [97] proposes recommendation-as-a-service platform for cloud configuration and pricing. It enables cloud users to explore various cloud configurations and recommends cloud services based on user preferences such as budget, performance expectation, and energy saving, etc. for a given workload and filter services based on the QoS requirements of the IaaS cloud service users.

This work considers only the IaaS domain and has considered only the QoS and not the QoE attributes of cloud services for recommendation. Some other works have also proposed recommendation engines for cloud services [98, 99, 100].

2.7 CHALLENGES IN MODELING RECOMMENDER SYSTEMS
Some of the important challenges [101,102] to be overcome in modeling a recommender system is given in Figure 2.4, Which includes the following: Data Scarcity, Scalability of the approach, Accuracy of prediction, Privacy concerns, Cold start problem, etc.

- **Data Scarcity Problem**

  Data Scarcity is a major problem in recommender systems developed for a new type of services. The data scarcity problem occurs when many users do not rate the items or services purchased and the user rating data available are scarce and predicting the user preferences becomes a challenge. Data Scarcity affects the accuracy of the predictions made by the recommender systems and in turn affects the quality of recommendations made to the users.

  The Collaborative recommender systems are the most affected systems due to data scarcity, since it solely depends on the rating given by the other similar users to
predict the item that may be of interest to the other users with similar taste. Many research works have focused on solving this problem. However, still more research works are needed in this area.

- **Scalability of the Approach**

  The scalability property of a recommender system is the ability of the system to handle the data growth over the time in a graceful manner. Many newly built recommender systems ignore this problem in the initial system building stage. However, when the system becomes popular and starts attracting more users, the scalability problem leads to inaccurate predictions and lower accuracy.

- **Accuracy of Prediction**

  Accuracy of prediction is another important research issue concerning the recommender systems. Many factors like the selecting the appropriate filtering method or choosing a combination of filtering methods based on the domain specific requirements plays a vital role in determining the accuracy of the prediction.

  Identifying different sources of reliable data, collecting and maintaining the historical, updated data about the items to be recommended is also a major challenge. There are many techniques and methods available in the literature to test the accuracy of the recommendations.

- **Privacy Concerns**

  Privacy concern is also one of the important research issues to be considered when building the recommender systems. As, the user data required by the filtering methods like the collaborative filtering are obtained from various sources like the social media, web logs, browser history, etc. the instances in which the privacy of a user can be breached is high, hence it is vital to address the privacy concerns of the users.
• **Over specialization problem**

   The over specialization problem is encountered in the recommender systems when the user is recommended frequently the items that are already defined or known through their profiles. The over specialization of recommended items, prevent the users from exploring other new items or options available to them.

• **Cold Start Problem**

   Many new types of domain specific services are evolving every day. Recommendation systems are essential to recommend the right services to the right user. It is a challenge, as recommender systems for a new type of services or items usually suffer from the cold start problem due to lack of data. In a cold start situation, predicting the right service for the user is difficult as enough data like the implicit and explicit user preferences, ratings and reviews about the new services are not available. The three types of common cold start situations are:

   a) Recommending new services to new users;
   b) Recommending existing services to new users;
   c) Recommending new services to existing users.

   While modeling the recommender systems for a new type of service, the cold start problem should be taken into consideration.

2.8 **COLD START PROBLEM IN RECOMMENDER SYSTEMS**

   Cold start problem is an important research issue in the field of a recommendation engine. Cold start occurs when the recommendation system could not predict the user preferences or interested items due to lack of information about the user and / or services. The cold start problem generally occurs in three different situations: i) Recommending new services to new users; ii) Recommending new services to existing users; and iii) Recommending existing services to new users. Many have worked towards overcoming this cold start problem in recommending items of
generic domains like music, movie, E-Commerce, Travel sites etc. The popular algorithm applied to overcome the cold start problem in recent researches is the collaborative filtering method.

Collaborative filtering method tries to identify the users with similar tastes and predicts the items to be recommended based on the similar user references. This method of recommendation is effective for recommending items for generic domains like music, movie, E-Commerce, Travel sites etc, since getting user preference data and identifying the users with similar tastes are relatively easy and data can be retrieved from various sources like social networks, blogs, search history of the users etc. However, it is not possible to apply the collaborative filtering methods to recommend technical services like the new type of cloud services, since the user requirements may vary for each specific project and it would be difficult to identify the users with similar tastes or requirements during the cold start situation.

### 2.8.1 Approaches to Solve Cold Start Problem in Recommender Systems

Many approaches have been provided in the literature to solve the cold start problem in recommender systems. The comparison of some of the important research works about solving the cold start problem is provided in the Table 2.4 and discussed in detail in this section.

Lucas Bernardi et al [103] have worked towards solving the Continuous Cold Start Problem that occurs in the travel site's recommendation domain. In this work, the Context aware collaborative filtering approach has been used, in which, the current context of the visitor and the behavior of other users in similar contexts are used to solve the cold start problem.

Jin-Hu Liu et al [104] have suggested methods to Promote items in Cold-Start situations in E-Commerce. In this work, the Top-k nearest neighbor algorithm has been used to perform the Item based Collaborative filtering.
A method to cold start problem in recommending the news items have been proposed by Michele Trevisiol et al [105]. In this work the topical filtering and the Content based Filtering methods have been used to overcome the cold start problem.

The approach of mining Large Streams of User Data for Personalized Recommendations has been explored by Xavier Amatriain et al [106]. In this work, they have applied the collaborative filtering techniques to predict the movie rating.

The work of Martin Saveski et al [107] Proposes to learn Local Collective Embeddings using the Multiplicative update rules based learning algorithm to solve the cold start problem in E-Commerce.

The Bayesian-inference Based Recommendation approach in Online Social Networks have been used by the Xiwang Yang et al [108] for overcoming the cold start problem. In their work, the users share their content ratings with friends and the similarity between a pair of friends is measured.

The approach proposed by Chien Chin Chen et al [109], Integrates a user model with trust and distrust networks to identify trustworthy users using the clustering method for movie recommendation. The method of using the Customers viewing behavior and the explicit user ratings given is used to predict the user preference for a TV program by Christopher Krauss et al [110].

The work of Katrien Verbert et al [111] explores information visualization techniques to interact with recommender systems and use collaborative filtering for recommending conferences and the work of Pankaj Gupta et al [112 ] uses graph recommendation algorithms for Twitter's user recommendation Service.

Jesus et al [113] have applied Jaccard similarity measures to generate recommendations for movies, where users have cast fewer votes. The Collaborative error-reflected models have been applied by N.M.Heung et al [114] for resolving the cold-start problem.

S.T.Park et al [115] has applied regression approach for cold start recommendations of movies. For recommending the scientific publications, S. Loh et al
has applied the Collaborative filtering methods and the Similarity function methods to overcome the cold start problem. Whereas, in the work of L. Martínez et al [117], the Collaborative filtering algorithm, knowledge-based filtering algorithm has been used to incomplete preference relations to smooth out the cold-start in collaborative recommender systems.

LT. Weng et al [118] suggests item taxonomies and item preference data for recommending books. The Cross-Level Association RulEs (CLARE) has been used to integrate the content into collaborative filters to recommend movies. Whereas, the work of S.T. C.W. Leung et al [119], has proposed a approach using the empirical study of a cross - level association rule mining to overcome the cold start problem. Park et al [120], analyses filterbots or surrogate users rating method to improve Collaborative Filtering algorithms.

Formal Concept Analysis (FCA) has been used for collaborative filtering by P.B. Ryan et al [121] for recommending movies. The content and collaborative data has been used under a single probabilistic framework for recommending movies by A. Schein et al [122].

From the techniques suggested above in the literature, it could be understood that most of the research works have used the collaborative filtering techniques in combination with other known techniques to overcome the cold start problem. This approach was effective as these research works concentrated on solving the cold start problem for general Business to Customer (B2C) type of E-Commerce domain.

However, these approaches could not be used for solving the cold start problem in Business to Business (B2B) type of E-Commerce domain like cloud services. This is because, these types of domain specific services, the data about the user preferences are scarce and the usually requires multiple criteria to be met for a service to be recommended. Also, the cost of losing a customer is high.
Table 2.4: Approaches to Overcome Cold Start Problem

<table>
<thead>
<tr>
<th>Author</th>
<th>Method / Algorithm Used</th>
<th>Application Domain</th>
<th>Limitations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lucas Bernardi et al [103]</td>
<td>Context aware collaborative filtering</td>
<td>Travel sites</td>
<td>Cold start problem in Business to Business (B2B) type of E-Commerce domain is not considered.</td>
</tr>
<tr>
<td>Jin-Hu Liu et al [104]</td>
<td>Top-k nearest neighbors</td>
<td>E-Com</td>
<td>Cold start problem in Business to Business (B2B) type of E-Commerce domain is not considered.</td>
</tr>
<tr>
<td>Michele Trevisiol et al [105]</td>
<td>Topical filtering, Content based Filtering methods</td>
<td>Online News</td>
<td>Data Scarcity issue is not considered.</td>
</tr>
<tr>
<td>Xavier Amatriain et al [106]</td>
<td>Collaborative filtering</td>
<td>Movie</td>
<td>Data Scarcity issue is not considered.</td>
</tr>
<tr>
<td>Martin Saveski et al [107]</td>
<td>Multiplicative update rules based learning algorithm</td>
<td>E-Com</td>
<td>Cold start problem in Business to Business (B2B) type of E-Commerce domain is not considered.</td>
</tr>
<tr>
<td>Xiwang Yang et al [108]</td>
<td>Bayesian-inference Method</td>
<td>Social Networks</td>
<td>Data Scarcity issue is not considered.</td>
</tr>
<tr>
<td>Chien Chin Chen et al [109]</td>
<td>Clustering Method</td>
<td>Movies</td>
<td>Data Scarcity issue is not considered.</td>
</tr>
<tr>
<td>Christopher Krauss et al [110]</td>
<td>Collaborative, Content based filtering, clustering Methods</td>
<td>TV Programs</td>
<td>Cold start problem in Business to Business (B2B) type of E-Commerce domain is not considered.</td>
</tr>
<tr>
<td>Pankaj Gupta</td>
<td>Graph recommendation</td>
<td>Social</td>
<td>Data Scarcity issue is not considered.</td>
</tr>
<tr>
<td>Author</td>
<td>Method / Algorithm Used</td>
<td>Application Domain</td>
<td>Limitations</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-------------------------</td>
<td>--------------------</td>
<td>----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>et al [112]</td>
<td>algorithms.</td>
<td>Networks</td>
<td>not considered.</td>
</tr>
<tr>
<td>Jesus et al [113]</td>
<td>Jaccard similarity measure</td>
<td>Movie</td>
<td>Data Scarcity issue is not considered.</td>
</tr>
<tr>
<td>N.M.Heung et al [114]</td>
<td>Collaborative filtering method.</td>
<td>Movie</td>
<td>Data Scarcity issue is not considered.</td>
</tr>
<tr>
<td>S.T.Park et al [115]</td>
<td>Regression on pairwise preference</td>
<td>Movie</td>
<td>Data Scarcity issue is not considered.</td>
</tr>
<tr>
<td>S. Loh et al [116]</td>
<td>Collaborative filtering methods, Similarity functions.</td>
<td>Scientific Articles</td>
<td>Quality of Experience (QoE) is not considered.</td>
</tr>
<tr>
<td>L. Martínez et al [117]</td>
<td>Collaborative filtering algorithm, knowledge-based filtering algorithm.</td>
<td>General</td>
<td>Cold start problem in Business to Business (B2B) type of E-Commerce domain is not considered.</td>
</tr>
<tr>
<td>LT. Weng et al [118]</td>
<td>Knowledge based filtering, Collaborative Filtering methods.</td>
<td>Books</td>
<td>Cold start problem in Business to Business (B2B) type of E-Commerce domain is not considered.</td>
</tr>
<tr>
<td>C.W. Leung et al [119]</td>
<td>Cross-Level Association RulEs (CLARE), Collaborative filtering.</td>
<td>Movie</td>
<td>Data Scarcity issue is not considered.</td>
</tr>
<tr>
<td>S.T. Park et al [120]</td>
<td>Filterbot algorithms</td>
<td>Movie</td>
<td>Data Scarcity issue is not considered.</td>
</tr>
<tr>
<td>P.B. Ryan et al [121]</td>
<td>FCA, Collaborative filtering.</td>
<td>Movie</td>
<td>Data Scarcity issue is not considered.</td>
</tr>
<tr>
<td>A. Schein et al [122]</td>
<td>Folding-in algorithm, Content based and Collaborative filtering.</td>
<td>Movie</td>
<td>Data Scarcity issue is not considered.</td>
</tr>
</tbody>
</table>
2.8.2 Cold Start Problem in recommending Cloud Services

Cold start problem in recommending the new type of cloud services is more complex to be solved than the cold start situations in recommending other general E-Commerce items. It is because, usually the cloud services selection and recommendation is based on the QoS requirements of the user and in the case of a new type of cloud service the monitoring and collection of the QoS data itself will be a challenge as the third party monitoring agencies may not be available to monitor the QoS of cloud services offered. Thus the appropriate QoS attributes that could be measured needs to be identified and the real time tests needs to be conducted to collect the QoS data about the services to be recommended.

Moreover, since these domain specific cloud services are mostly used by the business to business type of customers, collecting the data about the user preferences from other sources like the social media are also scarce. Hence, these issues in collecting and managing updated QoS, QoE data and also collecting other services related data like the new upgradation in the software version, cost, packages or offers, etc. makes the cold start problem of recommending the new domain specific cloud services very complex and challenging in all the three types of cold start situations.

2.9 DISCUSSION

Analyzing these recent works on recommending the cloud services, it is evident that most of the work has focused only on the Infrastructure-as-a-Service (IaaS) but has not explored the challenges involved in recommending domain specific Platform-as-a-Service (PaaS). Also, many other works which have focused on the ranking and selecting the cloud services have considered only the Quality of Service (QoS) aspects of the services and have not considered the Quality of Experience (QoE) of cloud services for ranking and selection of the IaaS type of services. Moreover, not many
have addressed the cold start problem in recommending new type of domain specific cloud services.

Though the work of Kailash et al [90] has addressed the cold start problem in recommending SaaS type of cloud services, the work has considered only the cloud settings for recommending general Software-as-a-Service (SaaS). However, even this work does not propose methods to overcome the core cold start problem persisting in recommending the domain specific services like cloud renderfarm services. The next chapter provides the details about the proposed methodology to overcome the cold start problem in recommending new type of cloud services like the cloud renderfarm services.